Building a Decentralized Voting System with Smart Contracts

#solidity#web3#tutorial#blockchain

Table of Contents

1. What are Smart Contracts?
2. Smart Contracts in Voting
3. Building a Voting System with Smart Contracts
4. Key Variables in Our Voting Smart Contract
5. Key Functions of Our Voting Smart Contract
6. Benefits and Challenges
7. Resources to Learn More
8. Conclusion

What are Smart Contracts?

Imagine a vending machine. You put in money, press a button, and get your snack. No human needed to make this transaction happen. That's the basic idea behind a smart contract!

A smart contract is like a digital vending machine. It's a computer program that automatically executes actions when certain conditions are met. Here's what makes them special:

1. **Automatic**: They run by themselves once they're set up.
2. **Trustless**: You don't need to trust a middleman. The code ensures everything happens as agreed.
3. **Transparent**: Everyone can see the contract's code and how it works.
4. **Secure**: They're stored on a blockchain, making them very hard to hack or change.

In simple terms, smart contracts are "if-then" statements on a blockchain. If something happens, then a specific action is taken.

Smart Contracts in Voting

Now, let's think about how we could use a smart contract for voting. It could work like this:

1. Set up the election (like setting up our vending machine)

2. Register voters (like giving people the right coins for our machine)
3. Cast votes (like pressing buttons on the machine)
4. Count votes and declare results (like the machine dispensing the most-chosen snack)

All of this happens automatically, securely, and transparently. No one can stuff the ballot box or change votes without everyone noticing.

Building a Voting System with Smart Contracts

Let's look at how we might build a simple voting system using a smart contract. We'll use a language called Solidity, which is commonly used for writing smart contracts.

```
// SPDX-License-Identifier: MIT

pragma solidity ^0.8.0;


contract VotingDApp {

  // Contract code will go here

}
```

This is the shell of our contract. Now, let's fill it with some useful parts.

Key Variables in Our Voting Smart Contract

Let's break down the main variables used in our contract:

```
address public owner;

uint256 public pollCount;

mapping(uint256 => Poll) public polls;

mapping(address => bool) public registeredCandidates;

mapping(uint256 => mapping(address => Candidate)) public pollCandidates;

mapping(uint256 => address[]) public allCandidates;
```

Here's what each of these does:

1. owner: This stores the Ethereum address of the person who created the contract. They have special permissions.

2. pollCount: This keeps track of how many polls have been created. It's like a counter.
3. polls: This is a way to store all the polls. Each poll has a number (its ID), and this mapping links that number to all the poll's information.
4. registeredCandidates: This keeps track of which Ethereum addresses have registered as candidates.
5. pollCandidates: This is a bit more complex. It stores information about candidates for each poll. You can look up a candidate by the poll ID and the candidate's address.
6. allCandidates: This stores a list of all candidate addresses for each poll.

These variables help us keep track of all the information we need for our voting system.

Key Functions of Our Voting Smart Contract

1. Creating a Poll

```
function createPoll(string memory name, string memory imageUrl, uint256 startTime, uint256 endTime) public onlyOwner {
    require(startTime < endTime, "Invalid poll duration");

    pollCount++;
    Poll storage poll = polls[pollCount];
    poll.name = name;
    poll.imageUrl = imageUrl;
    poll.active = false;
    poll.startTime = startTime;
    poll.endTime = endTime;

    emit PollCreated(pollCount, name, imageUrl, startTime, endTime);
}
```

This function is like setting up a new election. Here's what it does:

- Only the owner can create a poll (that's what onlyOwner means)
- It checks that the end time is after the start time
- It creates a new poll with a name, image, start time, and end time
- It tells everyone a new poll has been created (that's what emit does)

2. Registering as a Candidate

```
function registerAsCandidate(uint256 pollId, string memory candidateName, string memory imageUrl) public pollExists(pollId) {
    require(pollCandidates[pollId][msg.sender].candidateAddress == address(0), "Already registered as a candidate");
    require(!isCandidateNameTaken(pollId, candidateName), "Candidate name already taken");

    pollCandidates[pollId][msg.sender] = Candidate({
        candidateAddress: msg.sender,
        name: candidateName,
        imageUrl: imageUrl,
        approved: false,
        votesReceived: 0
    });

    allCandidates[pollId].push(msg.sender);

    emit CandidateRegistered(pollId, msg.sender, candidateName, imageUrl);
}
```

This function lets someone sign up to be a candidate:

- It checks that the poll exists and the person hasn't already registered
- It makes sure the candidate name isn't already taken

- It creates a new candidate with their name, image, and address
- It tells everyone a new candidate has registered

3. Voting

```
function vote(uint256 pollId, address candidate) public {
   Poll storage poll = polls[pollId];
   require(poll.active, "Poll is not active");
   require(!poll.hasVoted[msg.sender], "Already voted");
   require(pollCandidates[pollId][candidate].approved, "Candidate not approved");

   poll.hasVoted[msg.sender] = true;
   poll.votes[candidate]++;
   poll.totalVotes++;
   pollCandidates[pollId][candidate].votesReceived++;

   if (poll.votes[candidate] > poll.highestVotes) {
      poll.highestVotes = poll.votes[candidate];
      poll.winner = candidate;
   }

   emit VoteCasted(pollId, msg.sender, candidate);
}
```

This is the actual voting function:

- It checks that the poll is active and the voter hasn't already voted
- It makes sure the candidate is approved
- It records the vote and updates the vote count
- If this candidate now has the most votes, it updates the winner

- It tells everyone that a vote has been cast

4. Ending a Poll

```
function endPoll(uint256 pollId) public onlyOwner pollExists(pollId) {
    require(polls[pollId].active, "Poll is not active");

    polls[pollId].active = false;

    emit PollEnded(pollId, polls[pollId].winner);
}
```

This function ends a poll:

- Only the owner can end a poll
- It checks that the poll exists and is active
- It marks the poll as inactive
- It announces that the poll has ended and who the winner is

Benefits and Challenges

Benefits

- **Security**: It's very hard to cheat or hack the system.
- **Transparency**: Everyone can see how the voting works.
- **Automation**: No need for manual counting or management.
- **Accessibility**: People could potentially vote from anywhere.

Challenges

- **Technical Knowledge**: It can be complicated for some people to use.
- **Cost**: Running smart contracts on a blockchain can be expensive.
- **Scalability**: It might be hard to handle very large elections.
- **Privacy**: Balancing transparency with voter privacy can be tricky.

Resources to Learn More

If you're interested in learning more about smart contracts and blockchain technology, here are some great resources:

1. Ethereum.org's Introduction to Smart Contracts
2. Solidity Documentation
3. CryptoZombies - A fun, interactive way to learn Solidity
4. ConsenSys Academy - Offers courses on blockchain development
5. OpenZeppelin Learn - Tutorials and guides on smart contract development

Conclusion

Smart contracts offer an exciting new way to run voting systems. They can make voting more secure, transparent, and efficient. While there are challenges to overcome, the potential benefits make this an area worth exploring and developing further.

Remember, this is a simplified example. Real-world voting systems would need additional features and rigorous security measures. But hopefully, this gives you a good starting point for understanding how smart contracts can be used in voting!

You can find the full code for this voting system on GitHub: VotingDApp Smart Contract

Online Voting System using PHP Free Source Code

Submitted by jkev on Wednesday, June 30, 2021 - 14:37.

Language

PHP

Introduction of Online Voting System

This is a **PHP Project** is entitled **Online Voting System**. This was developed specifically for a **Secondary Level/High School** of **CHMSC**. The system handles the voting process of the school and this provides an online platform for the student to vote for their desired candidates.

About the System

The **Online Voting System** source code is written/built using **PHP, MySQL Database, HTML, CSS, Javascript, bootstrap**. The source code was originally developed using the older version of **PHP** and now upgraded to run properly in later versions such as **PHP7**. The system has 2 types of users which are the admin user and the voters. The admin user is in charge of managing the list of voters and candidates. The admin can also view and export to excel the canvassing report and the list of voters. The voters will have or be provided with an access or system credential to log in and vote for their desired candidates. The voters are only allowed to vote ones and the system will automatically reject the access if the voter is already voted to avoid duplication of votes.

Features

Admin

- **Login/Logout**
- **Manage Candidate**

- **Manage Voters**
- **Export Voters list to Excel**
- **View Canvassing Report**
- **Export Canvassing Report to Excel**
- **View System History log.**

Voters

- **Login/Logout**
- **Submit Vote**

The **Online Voting System** source code is free to download on this website. Feel free to download and modify the source code to enhance your knowledge of how the online voting system was developed and works.

How to Run

Requirements

- **Download** and **Install** any **local web server** such as **XAMPP/WAMP.**
- **Download** the provided source code **zip** file. (*download button is located below*)

Installation/Setup

1. **Open** your **XAMPP/WAMP's Control Panel** and start the **Apache** and **MySQL**.
2. **Extract** the **downloaded source code zip file**.
3. If you are using **XAMPP**, **copy** the extracted source code folder and **paste** it into the **XAMPP's "htdocs" directory**. And If you are using **WAMP**, **paste** it into the **"www" directory.**
4. **Browse** the **PHPMyAdmin** in a **browser**. i.e. **http://localhost/phpmyadmin**
5. **Create** a **new database** naming **ovs**.
6. **Import** the provided **SQL** file. The file is known as **ovs.sql** located inside the **db** folder.
7. **Browse** the Online Voting System in a **browser**.
   i.e. **http://localhost/Online_voting_system** for the voters
   and **http://localhost/Online_voting_system/admin** for the admin side.

Admin Access Information:

Username: **admin**
Password: **admin**

DEMO

That's it! I hope this **Online Voting System in PHP** will help you and you'll find something useful for your future **PHP Projects**.

Happy Coding :)

Note: Due to the size or complexity of this submission, the author has submitted it as a .zip file to shorten your download time. After downloading it, you will need a program like Winzip to decompress it.

Virus note: All files are scanned once-a-day by SourceCodester.com for viruses, but new viruses come out every day, so no prevention program can catch 100% of them.

FOR YOUR OWN SAFETY, PLEASE:

1. Re-scan downloaded files using your personal virus checker before using it.

2. NEVER, EVER run compiled files (.exe's, .ocx's, .dll's etc.)--only run source code.

Tags

php

voting system

eVoting

MySQL Database

Export to Excel

Bootstrap

Javascript

Comments

Submitted by Balakrishna Kini (not verified) on Fri, 11/16/2012 - 00:55

PAssword problem

I tried logining in after going to admin page but not able to login

- Reply

Submitted by jkev on Fri, 11/16/2012 - 10:10

In reply to PAssword problem by Balakrishna Kini (not verified)

thank you sir for the comment

thank you sir for the comment i try to fixed the problem thank you..

- Reply

Submitted by Balakrishna Kini (not verified) on Fri, 11/16/2012 - 16:01

In reply to thank you sir for the comment by jkev

You have already fixed the

You have already fixed the problem or you are yet to fix it ???

- Reply

Submitted by Anonymous (not verified) on Sat, 11/17/2012 - 13:24

In reply to You have already fixed the by Balakrishna Kini (not verified)

i try to download the program

i try to download the program but theres no error.. this is our project on our subject i prensent this to our class and theres no problem but i will try to fnd out what is your problem sir. thank you...

- Reply

Submitted by jonesfred on Sun, 12/02/2018 - 13:10

In reply to thank you sir for the comment by jkev

Customization

please i need customization to the voting system, email on jonesfred160@gmail.com

- Reply

Submitted by Aadill (not verified) on Sun, 12/15/2024 - 14:43

In reply to PAssword problem by Balakrishna Kini (not verified)

login error

incorrect username password popup

- Reply

Submitted by venner272000 on Mon, 12/10/2012 - 15:40

error

can u fix the error in login page please sir.. i encounter lots of problem and may i know what IDE did you use please i want to develop this project thanks in advance :)

- Reply

Submitted by carlo Bey (not verified) on Thu, 07/31/2014 - 16:35

In reply to error by venner272000

Login

My Problem is To develop The LoginForm Code

- Reply

Submitted by Anonymous (not verified) on Fri, 12/14/2012 - 11:02

voting system

everyone please try to explore on your own..i use wamp server and modify some codes and its working correctly..i am proud of the creator of this program,very nice indeed,the codes are simple and understandable...thank you so much

- Reply

Submitted by denndeal (not verified) on Mon, 02/17/2014 - 19:36

In reply to voting system by Anonymous (not verified)

help

wich codes did you modify???

- Reply

Submitted by Anonymous (not verified) on Mon, 03/20/2017 - 14:37

In reply to voting system by Anonymous (not verified)

What is the code that you've

What is the code that you've modify dude? I can't figure it out, please help.

- Reply

Submitted by nexus_00 on Sat, 01/05/2013 - 18:48

error login and variables undefined, callstack error

..please fix the bugs..every page that i link there is a php script error..also this web-application is not compatible to IE7 and other browser..try 2 make 8 cross-browser and cross-platform..use semantic codes..use also proper tabbing on your html and php srcipt..try to validate all the page with w3c validator..visit the w3.org.they have the validator for html and css..

- Reply

Submitted by Anonymous (not verified) on Mon, 01/07/2013 - 17:44

error

the system is nice but the log in and password is doesn't work.

- Reply

Submitted by Anonymous (not verified) on Mon, 02/25/2013 - 09:19

error login

Warning: session_start(): Cannot send session cache limiter - headers already sent (output started at C:\xampp\htdocs\votingsystem\admin\header.php:159) in C:\xampp\htdocs\votingsystem\admin\index.php on line 62 Warning: Cannot modify header information - headers already sent by (output started at C:\xampp\htdocs\votingsystem\admin\header.php:159) in C:\xampp\htdocs\votingsystem\admin\index.php on line 70

- Reply

Submitted by Anonymous (not verified) on Thu, 03/28/2013 - 20:17

ahhm medyo delay masyado ang

ahhm medyo delay masyado ang transaction..improve it...

- Reply

Submitted by aftabsoomro on Wed, 08/07/2013 - 21:32

errors in admin page

Warning: session_start(): Cannot send session cache limiter - headers already sent (output started at E:\xampp\htdocs\onlinevotingsystem\admin\header.php:159) in E:\xampp\htdocs\onlinevotingsystem\admin\index.php on line 62

- Reply

Submitted by ademaintain on Sun, 08/18/2013 - 18:36

TRY THIS

Warning: session_start(): Cannot send session cache limiter - headers already sent (output started at E:\xampp\htdocs\onlinevotingsystem\admin\header.php:159) in E:\xampp\htdocs\onlinevotingsystem\admin\index.php on line 62 There are so many causes of such error, buh for that page where the error is being called for, the user at his or her own end shuld bring out the connecting code, i.e., disable the include function....Need more help call +2348135567221

- Reply

Submitted by bishop gee on Thu, 01/07/2016 - 23:35

In reply to TRY THIS by ademaintain

help

i have a similar voting program.. i wanted to compare and contrast with his own.. only ryt nw i am very busy with other office and academic matters. if u hv sucessfully debugged d pgm pls cn u snd me a copy. thanks a lot

- Reply

Submitted by Mituram Ray (not verified) on Sat, 10/14/2017 - 22:38

In reply to TRY THIS by ademaintain

Just erase the include(

Just erase the include('session.php'); from index page :) :)

- Reply

Submitted by Nel (not verified) on Sat, 10/05/2013 - 15:38

Online Voting System

nice one! dinownload ko ung buong project, may mga konting kulang lang sa voting pero ok naman ung admin,naglagay din ako ng bar graph. :D salamat sa codes good work!

- Reply

Submitted by Anonymous (not verified) on Wed, 01/15/2014 - 19:17

sir asan po ung data base

sir asan po ung data base nito???

- Reply

Submitted by denndeal (not verified) on Fri, 03/14/2014 - 16:26

sql injection

the login pages of this codes are vulnerable to sql injection..can someone please correct it and post back?

- Reply

Submitted by philwainaina on Thu, 07/17/2014 - 19:07

why not working still

why not working still

- Reply

Submitted by Zidax on Wed, 09/20/2017 - 14:16

Empty Database

Hi can you provide or update the database with filled tables? the db you provided is empty

- Reply

Submitted by KF Sayson Durant on Tue, 05/21/2019 - 20:42

this is so good

this is so good

- Reply

Submitted by Limar (not verified) on Sun, 09/15/2019 - 10:17

IT

Sir Please i DOwnload this when i login it said no database selected

- Reply

Add new comment

Your name

Comment

Text format

Filtered HTMLFull HTML 1

About text formats

Subject

Save Preview

- Add new comment
- 34553 views